How to Perform Credible Verification, Validation, and Accreditation for Modeling and Simulation

Dr. David A. Cook and Dr. James M. Skinner
The AEgis Technologies Group, Inc.

Systems & Software Technology Conference

Monday, 18 April 2005
Track 4: 4:40 – 5:25 p.m.
Ballroom D

Many large-scale system development efforts use modeling and simulation (M&S) to lower their life-cycle costs and reduce risks. Unfortunately, these M&S tools can introduce new risks associated with potential errors in creating the model (programming errors) and inadequate fidelity (errors in accuracy when compared to real-world results). To ensure that a valid model and a credible simulation exist, verification and validation (V&V) of the model and the resulting simulation must be completed. This article discusses conducting effective and credible V&V on M&S.

According to the Department of Defense (DoD) "Online M&S Glossary," modeling and simulation (M&S) is defined as follows:

> The use of models, including emulators, prototypes, simulators, and stimulators, either statically or over time, to develop data as a basis for making managerial or technical decisions. [1]

While the terms *modeling* and *simulation* are often used interchangeably, adept practitioners of verification and validation (V&V) know and understand the difference.

A model is defined as "a physical, mathematical, or otherwise logical representation of a system, entity, phenomenon, or process." *Modeling* is the "application of a standard, rigorous, structured methodology to create and validate" this model [1]. Note that the implementation of a model is normally considered to be static; producing output from the model requires a simulation.

A simulation is defined as "a method for implementing a model over time." Separating the definition of the model from the simulation is an extremely useful method for developing analytical tools. This modular approach, combined with well-defined interfaces, allows you to update models as necessary without the need for updating the simulation software. It also supports a more thorough approach to V&V by allowing a V&V practitioner to separate the search for errors associated with the model from errors associated with the implementation of time. In fact, the certification of the results of an analytical study that relies on M&S tools requires that the model, the simulation, and the input data are all examined carefully prior to their use in supporting a decision.

The creation of a model, and its subsequent use in a simulation, is in itself a complex subject. Many good textbooks exist that serve as a guide to the creation of M&S. For a short but effective overview of M&S itself, refer to [2].

Introduction to V&V

For a moment, consider how critical V&V is to the DoD or any large organization that employs M&S tools. M&S tools are frequently used in acquisition or design decisions because: (1) the actual system has not been built yet, or (2) testing the actual system is too dangerous or cost-prohibitive. Since these decisions can involve billions of dollars, the safety of our troops, and the security of our nation, it is imperative that the credibility and limitations of M&S tools that we use to support our decisions be well understood. Therefore, prior to using an M&S tool, it should be subject to thorough V&V.

Verification is defined as:

> The process of determining that a model or simulation implementation accurately represents the developer's conceptual description and specification. Verification also evaluates the extent to which the model or simulation has been developed using sound and established software engineering techniques. [1]

In short, verification addresses the question "Have we built the model right?"

Validation is defined as:

> The process of determining the degree to which a model or simulation is an accurate representation of the real world from the perspective of the intended uses of the model or simulation. [1]

Validation considers the question "Have we built the right model?"

The two terms are often used together and incorrectly treated as if they were interchangeable. Two examples should help clear up any confusion. If a developer creates a model that accurately reflects the real-world system, but constantly crashes due to programming bugs, the system would fail verification but pass validation. A developer who correctly implements a bug-free program of a financial algorithm (provided by an expert) to predict the stock market would pass verification, but if the theory used by the expert was flawed, the model would fail validation.

Another common misconception is that V&V is synonymous with testing. V&V does not replace testing, nor does it include testing – instead, V&V, when used properly, can determine if testing has been performed correctly. Testing is an important activity in all software life cycles. V&V, while not normally a life-cycle activity, makes sure that all life-cycle activities have been correctly performed – including testing. The V&V of M&S is also different from V&V of other software artifacts. In M&S, the V&V is used to show that the software model is a useful representation of the real world.

An important part of V&V that is often overlooked is the certification of the input data and default values that the M&S tool relies on for a study. Often, developers will claim that they are not responsible for the input values, placing the responsibility instead on the analyst that uses the tool. The claim is that there is no problem with the model, and that everyone understands that any software suffers from *garbage in, garbage out*.

In truth, when analysts first receive a new tool, they normally assume that the developer has more experience than they have with the software, and will accept default values unless they have a specific reason to change them. Therefore, when conducting V&V on a tool, it is necessary to question the source of every default value in the system. The developer cannot be held responsible if the values are not exact, but they should be held responsible for providing reasonable default values. In our opinion, the most common problem you will find is that the default value of zero is less likely to be *a reasonable guess* and more likely to be a placeholder simply because the developer did not know.

Informal	Static	Dynamic		Formal
Audit	Cause-Effect Graphing	Acceptance Testing	Predictive Validation	Induction
Desk Checking	Control Analysis	Alpha Testing	Product Testing	Inference
Face Validation	• Calling Structure	Assertion Checking	Regression Testing	Logical Deduction
Inspections	• Concurrent Process	Beta Testing	Sensitivity Analysis	Inductive Assertions
Reviews	• Control Flow	Bottom-Up Testing	Special Input Testing	Lambda Calculus
Touring Test	• State Transition	Comparison Testing	• Boundary Value	Predicate Calculus
Walkthroughs	Data Analysis	Compliance Testing	• Equivalence Partitioning	Predicate Transformation
	Data Dependency	• Authorization	• Extreme Input	Proof of Correctness
	Data Flow	• Performance	• Invalid Input	
	Fault-Failure Analysis	• Security	• Real-Time Input	
	Interface Analysis	• Standards	• Self-Driven Input	
	• Model Interface	• Debugging	• Stress	
	• User Interface	Execution Testing	• Trace-Driven Input	
	Semantic Analysis	• Monitoring	Statistical Techniques	
	Structural Analysis	• Profiling	Structural (White Box) Testing	
	Symbolic Evaluation	• Tracing	• Branch	
	Syntax Analysis	Fault/Failure Insertion Testing	• Condition	
	Traceability Assessment	Field Testing	• Data Flow	
		Functional (Black Box) Testing	• Loop	
		Graphical Comparisons	• Path	
		Interface Testing	• Statement	
		• Data	Submodel/Module Testing	
		• Model	Symbolic Debugging	
		• User	Top-Down Testing	
		Object-Flow Testing	Visualization/Animation	
		Partition Testing		

Table 1: *Possible V&V Techniques [4]*

Why M&S Has Increased V&V Needs

Developing an M&S application is not significantly different from any other software application. However, certain types of M&S applications have more stringent V&V needs. M&S is typically used for one of three purposes: descriptive, predictive, and normative models. Descriptive models are intended to provide a characterization of the nature and workings of the modeled process – to explain how a real-world activity functions. Predictive models, usually more complex than descriptive models, are designed to predict future events in addition to describing objectives and events. Normative (or control) models are the most difficult and complex models to construct since these models not only describe and predict, but also provide direction about the proper course of action.

Descriptive models require V&V just like any other software activity. Typical software has an output that, once verified and validated, can be used. Predictive and normative models require additional V&V because the output of these models is used to predict and guide future actions, often without a human in the loop. Because of the potentially high cost of failure, V&V is critical for these types of models. Because of this, separate and additional V&V for the M&S is frequently merited.

It is no secret that requirements are an integral part of all software activities. In fact, requirements engineering is fundamental to developing useful and valid software. Nowhere are requirements more important than in M&S. Prior to attempting to use M&S to help save time or costs in your system, make sure that a mature requirements engineering program is in place [3].

Possible V&V Techniques

Once the decision has been made to use M&S – and, of course, V&V of the M&S tools – you will need to create a V&V plan detailing which activities will provide you with the highest level of confidence in the tools. A multitude of different V&V techniques exist that have been derived from software engineering and statistical methods. The best starting point for a newly appointed V&V agent to learn about these techniques is to obtain a copy of the "Recommended Practices Guide" (RPG) published by the Defense Modeling and Simulation Office (DMSO) [4]. Table 1 lists over 75 different techniques described in the RPG.

DMSO provides this list as a set of tools from which a practitioner can select the most appropriate techniques for their particular project. It is not necessary, or even advisable to attempt to apply all of the techniques to any individual project. Many of the techniques are overlapping in their coverage, and it requires experience to determine which technique is the best to meet a project's needs. To understand this more clearly, consider the four major categories into which the RPG divides the techniques: informal, static, dynamic, and formal. The following paragraphs are not designed to fully explain or cover the four techniques. Instead, they offer insights into the rationale behind the major categories.

Informal V&V Techniques

Do not let the term *informal* mislead you. While informal techniques have the advantage that they are relatively easy to perform and understand, their application is anything but unstructured. In fact, several of the methods such as desk checking (also known as *self-inspection*) can have very detailed checklists. Watts Humphrey, among others, gives techniques for developing highly effective checklists that can be used as part of a very rigorous personal review [5]. Informal V&V techniques can be very effective if applied with structure and guidelines, and they are relatively low cost. Informal V&V techniques are effective for examining both the model and the simulation.

Static V&V Techniques

Static V&V techniques are basically all of the activities that can be performed without executing the code. These techniques are used almost exclusively to examine the model and its implementation. Unfortunately, static V&V techniques do not examine the execution of the model, and therefore they are of limited usefulness in M&S V&V. Static V&V techniques can be used on the code of the model as it is being developed, but static techniques are not effective on simulations themselves, as simulations require execution of the model. This is not to say that static techniques are not useful in M&S; instead, we are saying that static techniques only perform V&V on the model, and ignore the simulation. During M&S V&V, you must ensure that dynamic techniques are also used for simulation V&V.

Dynamic V&V Techniques

Dynamic V&V techniques look at the results of the execution of the model. At the simplest level, dynamic V&V can be merely examining the output of an execution. However, that is almost always insufficient. Instead, the model must be examined as it is being executed. This typically requires *instrumenting* – the insertion of additional code into the model to collect or monitor model behavior during execution. Normally, the steps involved are to instrument the model with V&V code, execute the model, and then analyze the dynamic behavior and output of the model.

While these are extremely useful techniques, instrumenting a model changes it slightly. To observe the dynamic execution of a model requires additional instructions to collect data. These additional instructions can slightly modify the timing or behavior of the model. A dictum to remember in dynamic V&V is, "Those who observe, perturb!" Great care must be used in instrumenting simulation code to ensure that the instrumentation itself does not affect the validity of the simulation output
.

Formal V&V Techniques

Formal V&V techniques rely on formal mathematical reasoning, inference, and proofs of correctness. While these are effective means of V&V, they are often very costly. The difficulty lies in both the complexity of the techniques and the size of the model under examination. Many formal techniques – while extremely effective – are unusable for other than trivial simulations. Others require an understanding of complex mathematics – skills not common in most developers.

While formal techniques may not be practical for most models and simulations, some of the basic concepts of the formal methods are used in other techniques (assertions, pre- and post-conditions, etc.). Automated M&S development tools of the future have the potential for designing and implementing M&S with formal methods. However, based on the authors' experience, at the current time formal methods are infrequently used.

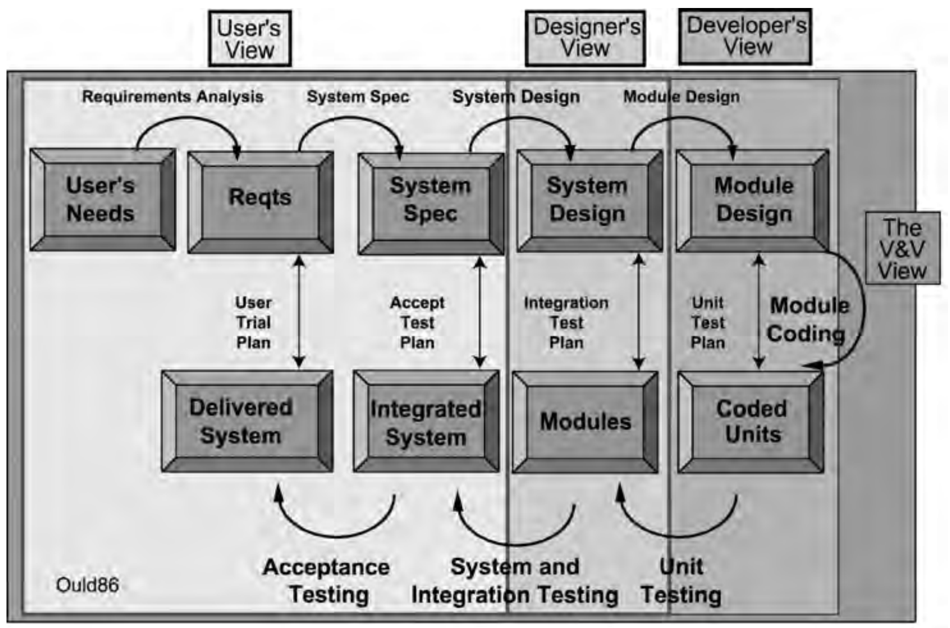
Life-Cycle Activities

The four categories of V&V techniques provide the basic tools for performing V&V. However, they only suggest which techniques are available. Determining how and when to apply the techniques requires judgment. In many organizations, V&V is performed by outside agents with the experience to evaluate the current program and suggest cost-effective ways to effectively apply a selected subset of the techniques listed in Table 1. Even when internal resources are used for M&S V&V, it is important to consider how V&V activities relate to the entire M&S development life cycle. The techniques you use for V&V are not as important as making sure you cover all software life-cycle development steps.

While many references explaining the software life cycle exist, we have found that the life-cycle diagram found in [6], shown in Figure 1, helps developers in understanding the typical life cycle of an M&S application.

Figure 1 shows four important viewpoints: the user, the designer, the developer, and the V&V views. The rectangles in this figure represent products associated with the project; the arcs are the processes that translate one product into a subsequent product with a different viewpoint and possibly a different audience. Following the path of the arcs shows how the view of the user (captured in a user's needs document such as a Statement of Need) is translated first to the designer's view and the developer's view, and then, as it is tested, is returned ultimately to the user's view as a delivered system. However, while the user, designer, and developer only need to be concerned with a slice of the process, the V&V agent needs to have insight into the entire software development life cycle.

Figure 1: *A Mature Software Development Life Cycle for M&S*



The number and selection of V&V activities that should be conducted on an M&S tool depend in part on the purpose of the tool. As stated earlier, M&S tools can generally be categorized as being one of three types: descriptive, predictive, and normative. Descriptive models are intended to provide a characterization of the nature and workings of the modeled process. Predictive models are usually more complex; in addition to describing objects and events, they are designed to predict future events. Normative (or control) models are the most difficult models to construct since these models not only describe and predict, but provide direction about the proper course of action.

As you might imagine, predictive and normative M&S tools warrant a larger set of V&V activities than required by a descriptive M&S tool, particularly in the area of requirements engineering. Since these tools are intended to predict future events, a mechanism is needed to capture changing environments to ensure that even after a tool is delivered, changes in the environment are captured and used to refine the design of the system. In fact, as in all software programs, requirements engineering is fundamental to developing useful and valid M&S tools. Prior to attempting to use M&S to help save time or costs in your system, make sure that a mature requirements engineering program is in place [4].

Sample Activities

Selecting from the more than 75 techniques can be intimidating, especially for a new practitioner. The key is to realize that V&V is most effective when the selection of techniques covers the entire life cycle of the model. By focusing on the life cycle – with the goal of ensuring that each of the steps of the development process was performed adequately – a V&V agent can select a logical subset of techniques that is reasonable, sufficient, and affordable.

Figure 2 depicts a sample set of activities that provides adequate coverage of the life cycle. These activities are all found in Table 1, and each activity is associated with a specific life-cycle activity. Two of the activities, Formal Document Review and Inspection of Configuration Management Practices, are not associated with any specific life-cycle step, but encompass the entire life cycle.

Formal document review and an inspection of configuration management practices were added in order to gauge the maturity of the developing organization. It has been our experience that organizations with Capability Maturity Model® Level 2 processes in place will have documented requirements, designs, and test results that contribute greatly to the confidence in the final product. In addition, we perform a thorough V&V of all input and default values used by the program by requiring the developer to cite a source for every value used by the program. A relatively inexpensive and quick final sanity check is to have subject matter experts (SME) conduct face validation in which the SMEs simply compare the simulation results to their experience and expectation and tell us if it *looks right*.

When constructing a V&V plan for an organization, the steps shown in Figure 2 are tailored and supplemented as warranted by the specific needs of the end user. A report is prepared that provides details for each activity conducted, including the following:

- The description and process of the technique selected.
- The unit under test (that is, the artifact examined, for example, code, documentation, Software Requirement Specification).
- Results.
- Conclusions.
- Rating.
- Recommendations.

The rating system used is relatively simple – we assign the rating green, yellow, or red, signifying no significant problems, concerns or limitations, or serious discrepancies, respectively. This relatively simply rating system (rather than a simply pass fail, or a more complex numeric rating system) gives insight into the results of the V&V activities without the necessity of creating a complex scoring system.

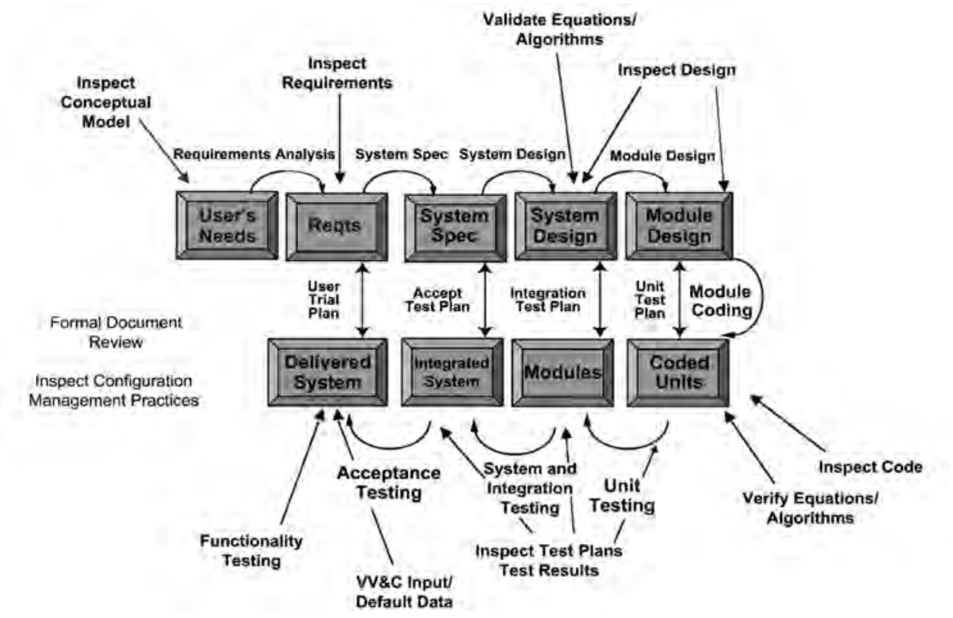


Figure 2: *Sample V&V Activities That Span the M&S Life Cycle*

Accreditation

Frequently, after a model and simulation have been thoroughly tested and V&V have been accomplished, we are asked to make a recommendation as to accreditation. Accreditation is a complex topic in itself and is defined as, "The official certification that a model or simulation is acceptable for use for a specific purpose" [1]. In essence, an accreditation assessment results in a recommendation that the certifying official should authorize that the M&S tool can be used for the purpose it has been designed for: descriptive, predictive, or normative. Not all M&S projects need an accreditation, so this step is performed only if necessary.

By examining the requirements of the model and simulation, and by examining the purpose that the model was designed for, V&V permits you to recommend how (or if) it should be used. There are five possible accreditation recommendations:

1. The model or simulation will be used *as described*.
2. The model or simulation will be used as described *with limitations*.
3. The model or simulation will be used as described *with modifications*.
4. The model or simulation *requires additional V&V* to be considered suitable for accreditation.
5. The model or simulation *will not be used* for this application.

Most models we examine are given recommendation No. 2 – accredit with limitations. Almost all models have limitations, which is why, during V&V activities, it is important to document any limitations.

Conclusions

M&S is an effective means of lowering life-cycle costs and can shorten development time and prevent the construction of the final product until many *what-if* questions have been answered. However, unless the requirements are valid, the resulting model (and simulation output) will be useless. Even if the requirements are valid, the results of the simulation will not be trustworthy unless care is taken during construction of the model and during the execution of the simulation.

To guarantee that you have a valid model and simulation that produce correct results, V&V of both the model and the simulation must be accomplished. There are a wide variety of V&V techniques to choose from. V&V techniques of the model and simulation, however, are not adequate by themselves. Along with V&V techniques, you must also perform V&V of the accompanying life-cycle steps used in the construction of the M&S. To sum it up, V&V is required to know that you have the right model and valid simulation results that you can trust.◆

References

1. Defense Modeling and Simulation Office. "Online M&S Glossary." DoD 5000.59M. Washington: Department of Defense <www.dmso.mil/public/resources/glossary>.
2. Cook, David A. "Computers and M&S - Modeling and Simulation 101." CROSSTALK Jan. 2001 <www.stsc.hill.af.mil/crosstalk/2001/01/cook.html>.
3. VanBuren, Jim, and David A. Cook. "Experiences in the Adoption of Requirements Engineering Technologies." CROSSTALK Dec. 1998 <www.stsc.hill.af.mil/crosstalk/1998/12/index.html>.
4. Defense Modeling and Simulation Office. "Recommended Practices Guide." Washington: Department of Defense <http://vva.dmso.mil>.
5. Humphrey, Watts. A Discipline for Software Engineering. Addison-Wesley, 1995.
6. Ould, Martyn A., and Charles Unwin. Testing in Software Development. Press Syndicate of the University of Cambridge, 1986.

Coming Events

June 4-8
The 32nd Annual International Symposium on Computer Architecture
Madison, WI
http://www.cs.wisc.edu/~isca2005

June 6-9
SUPERCOMM 2005
Chicago, IL
http://www.supercomm2005.com

June 12-15
ACM Sigplan 2005 Programming Language Design and Implementation
Chicago, IL
http://research.ihost.com/pldi2005

June 15-17
LCTES '05
Conference on Languages, Compilers, and Tools for Embedded Systems
Chicago, IL
http://soarlab.snu.ac.kr

June 15-17
6th IEEE Information Assurance Workshop
West Point, New York
www.itoc.usma.edu/workshop/2005

June 20-23
2005 World Congress in Applied Computing Conference
Las Vegas, NV
www.world-academy-of-science.org/WCAC2005/ws

June 28-30
IADC 2005
International Advanced Database Conference
San Diego, CA
www.conferencehome.com/iadc.htm

May 1-4, 2006
2006 Systems and Software Technology Conference

Systems & Software Technology Conference

Salt Lake City, UT
www.stc-online.org

About the Authors

David A. Cook, Ph.D., is a senior research scientist at AEgis Technologies Group, Inc., working as a Verification, Validation, and Accreditation agent in the Modeling and Simulations area. He is currently supporting verification, validation, and accreditation for the Missile Defense Agency Airborne Laser program. Cook has more than 30 years experience in software development and software management. He was formerly an associate professor of computer science at the U.S. Air Force Academy, and was a former deputy department head of the Software Professional Development Program at the Air Force Institute of Technology. He was a consultant for the U.S. Air Force Software Technology Support Center for more than six years. He has a doctorate in computer science from Texas A&M University, and is an authorized Personal Software Process instructor.

The AEgis Technologies Group, Inc.
6565 Americas PKWY
STE 975
Albuquerque, NM 87110
Phone: (505) 881-1003
Fax: (505) 881-5003
E-mail: dcook@aegistg.com

James M. Skinner, Ph.D., is a senior research scientist at AEgis Technologies Group, Inc., with more than 20 years of research experience. Prior to joining AEgis, Skinner served in the U.S. Air Force for 20 years, primarily in research and academic environments. His assignments included research positions at the Air Force Research Laboratory, Sandia National Laboratory, and the Air Force Institute of Technology. Skinner currently manages modeling and simulation, and verification, validation, and accreditation projects in support of the Airborne Laser Program. He has a Bachelor of Science in electrical engineering from the University of Washington, a Master of Science in computer engineering from the Air Force Institute of Technology, and a doctorate degree in computer science from the University of New Mexico.

The AEgis Technologies Group, Inc.
6565 Americas PKWY
STE 975
Albuquerque, NM 87110
Phone: (505) 881-1003
Fax: (505) 881-5003
E-mail: jskinner@aegistg.com

Online Article

Automated Restructuring of Component-Based Software

Systems & Software Technology Conference
Thursday, 21 April 2005
Track 7: 12:25 - 1:10 p.m.
Room 251 A-C

Robert L. Akers, Ira D. Baxter, and Michael Mehlich, *Semantic Designs;* Brian Ellis and Kenn Luecke, *The Boeing Company*

Reengineering legacy software to use a modern component model can be accomplished by repeatedly applying a large number of semantically sensitive program transformations, some of which synthesize new code structures, while others modify legacy code and meld it into the new framework. Using machinery that automates the process conquers the problems of massive scale, soundness, and regularity, and furthermore reduces time to completion by overlapping the project's design and implementation phases. This article describes experience in automating the reengineering of a collection of avionics components to conform to a CORBA-like component framework, using Design Maintenance System, a program analysis and transformation system designed for building software engineering tools for large systems.

CrossTalk is excited to offer this additional article. For the text of this article, go to <www.stsc.hill.af.mil/crosstalk/2005/05/0505Akers.html>.